

**DETECTING AND HANDLING AFFINITY
BREAKS IN WEB APPLICATIONS**

5

ABSTRACT OF THE DISCLOSURE

A system and method are disclosed, according to which, the responsiveness of client/server-based distributed web applications operating in an object-oriented environment may 10 be improved by detecting and handling affinity breaks between a client and server. Affinity exists when a client's requests are all routed to the same server. This is often the case, for example, during secure online transactions. A problem occurs when the preferred server 15 becomes temporarily unavailable, and the client's requests are directed to a different server. When the original server is restored, it must detect the fact that its affinity with the client was disrupted, and any client-specific data in its cache may be invalid. In an 20 exemplary embodiment of the system and method disclosed herein, an "affinity command" is exchanged between the client and the server during each client request. The affinity command contains a user ID associated with the client and a "generation ID", which is unique for each 25 request. When a request is received from a client, the server examines the generation ID in the accompanying affinity command and compares it to its internally recorded value. If they match, the server knows it has not missed any requests. In this case, the server updates the 30 generation ID and sends the new value back to the client, along with the requested cache entry. If the generation ID received from the client fails to match the recorded value, an affinity break is detected, and the server updates its cache from the database.